# Algorithmic Evaluation in Modern Hockey: A Technical Deconstruction of Probabilistic Scoring, Regularized Regression, and Spatial Tracking

## 1. Epistemological Foundations of Stochastic Game Analysis

The transition of National Hockey League (NHL) evaluation from frequency-based counting statistics to probabilistic modeling represents a fundamental shift in how the sport is quantified. Traditional metrics—Goals, Assists, and Plus-Minus—are deterministic descriptions of outcomes, but they suffer from significant variance and fail to isolate individual contributions from the systemic noise of linemates, competition, and usage. The objective of modern hockey analytics is to strip away this contextual noise to reveal the latent "true talent" of a player. This report provides an exhaustive technical analysis of the three primary mechanisms used to achieve this: **Expected Goals (xG)** for assessing shot quality, **Regularized Adjusted Plus-Minus (RAPM)** for isolating player impact via ridge regression, and **Wins Above Replacement (WAR)** for aggregate value assessment. Furthermore, it explores the emerging frontier of **Computer Vision and Tracking Data**, specifically examining micro-statistics such as Gap Control and Faceoff Valuation (WDBE).

To replicate the sophisticated models utilized by industry leaders such as Evolving Hockey, MoneyPuck, and TopDownHockey, researchers must navigate a complex pipeline of data engineering, feature extraction, and machine learning. This report details the specific mathematical formulas, data structures, and algorithmic choices required to reconstruct these frameworks from raw NHL API data.

## 2. Data Engineering Architecture and Coordinate Geometry

The prerequisite for any advanced modeling in hockey is the construction of a robust data pipeline capable of ingesting semi-structured JSON data and normalizing it into a spatial-temporal format suitable for regression analysis. The NHL's shift to the EDGE (Puck and Player Tracking) API has introduced new complexity, requiring the fusion of event-based "Play-by-Play" (PBP) data with high-frequency tracking feeds.

### 2.1 API Endpoint Taxonomy and JSON Structure

Replication begins with data acquisition. The NHL API ecosystem consists of distinct endpoint classes that must be joined to create a complete analytical picture.

#### 2.1.1 The Play-by-Play (PBP) Event Feed

The foundational dataset for xG and RAPM is the PBP feed, accessible via the gamecenter endpoints. This data provides a discrete log of game events. To replicate modern models, the following JSON structure must be parsed and flattened 1:

* **Endpoint:** https://api-web.nhle.com/v1/gamecenter/{game\_id}/play-by-play
* **Key JSON Objects:**
  + plays: An array of event objects.
  + details.xCoord / details.yCoord: The spatial location of the event.
  + typeDescKey: The event classification (e.g., SHOT\_ON\_GOAL, FACEOFF, HIT).
  + situationCode: A 4-digit code representing the goaltender/skater state (essential for determining strength state like 5v5 or 5v4).

#### 2.1.2 The EDGE Tracking Feed

For micro-statistics (Gap Control, Burst Speed), the PBP data is insufficient. The EDGE API provides telemetry data at approximately 25 Hz.

* **Endpoint:** https://api-web.nhle.com/v1/gamecenter/{game\_id}/cayenne/points (and related skater-data endpoints).
* **Data Payload:** Contains time-series data for every player on the ice, including x, y coordinates, velocity, and acceleration vectors.1

### 2.2 Coordinate Normalization and Rink Standardization

A critical failure point in early replication attempts is the mishandling of rink geometry. The NHL defines the rink with coordinates $(0,0)$ at center ice, extending to $x = \pm 100$ ft and $y = \pm 42.5$ ft. However, raw data records coordinates relative to the arena's fixed camera, meaning a team attacks positive $x$ in periods 1 and 3, and negative $x$ in period 2.4

The Normalization Algorithm:

For spatial models like xG, all shots must be mathematically transformed to a single "offensive half" perspective (typically mapping the offensive net to $x=89, y=0$).

1. **Identify Possession:** Determine the "Event Owner" (the team taking the shot).
2. **Determine Rink Side:** Analyze the distribution of the team's events for the current period. If the median $x$ of their shots is negative, they are attacking the "left" side.
3. Transformation Logic:  
   If the team is attacking the left side (negative $x$):  
     
   $$x\_{norm} = -1 \cdot x\_{raw}$$  
   $$y\_{norm} = -1 \cdot y\_{raw}$$  
     
   This ensures that the distance calculation is consistent regardless of the period.4

### 2.3 The "Stint" Data Structure for Regression

While xG models operate on *events* (rows = shots), RAPM models operate on *stints* (rows = time intervals). A "stint" is defined as a continuous duration of gameplay where the ten skaters and two goaltenders on the ice remain constant.6

Replication Requirement:

To build the RAPM design matrix, one must iterate through the PBP feed and trigger a new row generation whenever:

1. A player substitution occurs.
2. A goal is scored.
3. A penalty begins or ends.
4. A period ends.

**Table 1: Stint-Level Data Dictionary for RAPM Construction**

| **Variable Name** | **Data Type** | **Description** | **Usage in Regression** |
| --- | --- | --- | --- |
| stint\_id | Integer | Unique key for the time interval | Row Identifier |
| duration | Float | Length of stint in seconds | **Sample Weight ($w$)** |
| home\_players | List[Int] | IDs of 6 home players | Feature Engineering (1) |
| away\_players | List[Int] | IDs of 6 away players | Feature Engineering (-1) |
| score\_state | Categorical | Score differential (e.g., +1, -2) | Contextual Control |
| zone\_start | Categorical | Location of preceding faceoff | Contextual Control |
| GF | Integer | Goals For during this stint | Target Variable ($Y$) |
| xGF | Float | Expected Goals accumulated in stint | Target Variable ($Y$) |

## 3. The Predictive Engine: Expected Goals (xG) Methodology

Expected Goals (xG) is a classification framework that assigns a probability $p \in $ to every unblocked shot attempt, representing the likelihood of that shot becoming a goal. While early iterations used Logistic Regression, the current industry standard—exemplified by MoneyPuck and Evolving Hockey—utilizes **Gradient Boosting Machines (GBM)**, specifically implementations like XGBoost or LightGBM.8

### 3.1 Feature Engineering: The "Royal Road" and Geometric Logic

The predictive power of an xG model is derived almost entirely from feature engineering. The most significant advancement in recent literature is the quantification of pre-shot movement, specifically the "Royal Road" concept.

#### 3.1.1 The Royal Road Calculation

The "Royal Road" is the imaginary line connecting the two nets through the center of the ice ($y=0$). A pass that crosses this line immediately prior to a shot forces the goaltender to reset their angle and move laterally, drastically increasing scoring probability.11

To replicate this feature computationally:

1. **Retrieve Previous Event:** Identify $E\_{t-1}$ (pass/event) and $E\_t$ (shot).
2. Boolean Logic:  
     
   $$RoyalRoad = \begin{cases} 1 & \text{if } \text{sign}(y\_{t-1}) \neq \text{sign}(y\_{t}) \text{ AND } \Delta t < 3.0s \\ 0 & \text{otherwise} \end{cases}$$
3. **Angle Change:** A continuous variable representing the absolute difference in shot angle between the puck's location at $t-1$ and $t$.

#### 3.1.2 Geometric Features

Standard distance is insufficient. Replicating MoneyPuck's model requires specific derivations:

* Shot Distance ($d$): Euclidean distance to the net center ($89, 0$).  
    
  $$d = \sqrt{(89 - x\_{norm})^2 + (y\_{norm})^2}$$
* Shot Angle ($\theta$): The absolute angle from the central axis.  
    
  $$\theta = \left| \arctan \left( \frac{y\_{norm}}{89 - x\_{norm}} \right) \right| \times \frac{180}{\pi}$$
* Speed from Previous Event: A proxy for game speed and puck velocity.  
    
  $$Speed = \frac{\sqrt{(x\_t - x\_{t-1})^2 + (y\_t - y\_{t-1})^2}}{time\\_elapsed}$$  
    
  Note: This feature effectively captures one-timers and rebounds without needing explicit video tagging.8

### 3.2 The Flurry Adjustment Algorithm

A major flaw in naive xG summation is the "Flurry Effect." If a player takes three shots in two seconds (a shot and two rebounds), a raw sum might yield an xG of $0.4 + 0.5 + 0.6 = 1.5$. This is probabilistically incoherent, as a single possession cannot yield more than one goal.

MoneyPuck's Correction Formula:

To replicate the "Flurry Adjusted xG," one must calculate the probability of not scoring in the sequence and invert it. For a sequence of $n$ related shots:

$$xG\_{flurry} = xG\_{1} + \sum\_{i=2}^{n} \left( xG\_{i} \times \prod\_{j=1}^{i-1} (1 - xG\_{j}) \right)$$Alternatively, interpreted as the total event probability:

$$P(AtLeastOneGoal) = 1 - \prod\_{i=1}^{n} (1 - xG\_i)$$

In replication, shots are grouped into a "flurry" if they occur within same stoppages and are taken by the same team.14

### 3.3 Shooting Talent Adjustment (Bayesian Inference)

Standard xG models are "shooter-blind," meaning an Ovechkin shot is valued the same as a fourth-liner's shot. To create "Shooting Talent Adjusted xG," researchers apply a Bayesian prior to the residuals ($Goals - xGoals$).

**Replication Logic:**

1. Calculate raw residual: $R = Goals - xG\_{raw}$.
2. Apply Regressed Shooting Talent ($ST$):  
     
   $$ST = \frac{R \cdot \text{Shots}}{\text{Shots} + \text{Reg\\_Threshold}}$$  
     
   where $\text{Reg\\_Threshold}$ represents the number of shots required to regress the result halfway to the mean (typically ~300-500 shots).
3. Adjusted xG: $xG\_{talent} = xG\_{raw} + (ST / \text{Shots})$.  
   This method allows the model to credit elite finishers without overfitting to small sample variance.15

## 4. The Isolating Engine: Regularized Adjusted Plus-Minus (RAPM)

While xG evaluates *events*, RAPM evaluates *players*. It is the industry standard for isolating a player's contribution from the confounding variables of teammates, quality of competition, zone starts, and schedule fatigue. The mathematical engine driving RAPM is **Ridge Regression** ($L\_2$ Regularization).6

### 4.1 The Multicollinearity Problem and the Ridge Solution

In a linear system $Y = X\beta + \epsilon$, the matrix $X$ contains dummy variables for every player. In hockey, players shift in units (lines and pairings), causing extreme multicollinearity. If Player A and Player B always play together, Ordinary Least Squares (OLS) cannot mathematically distinguish their coefficients, leading to variance inflation.

Ridge Regression Formula:

To solve this, RAPM minimizes a penalized sum of squares. The objective function is:

$$\hat{\beta}\_{ridge} = \underset{\beta}{\text{argmin}} \left( \sum\_{i=1}^{N} w\_i (y\_i - x\_i^T \beta)^2 + \lambda \sum\_{j=1}^{p} \beta\_j^2 \right)$$

* **$y\_i$:** The target rate (e.g., GF/60) for stint $i$.
* **$x\_i$:** The vector of predictors for stint $i$ (players, zone, score).
* **$w\_i$:** The weight (duration) of stint $i$.
* **$\lambda$:** The regularization parameter (penalty term).

The $\lambda \sum \beta^2$ term forces coefficients toward zero, "shrinking" the variance and allowing the model to assign unique values to linemates based on the rare minutes they play apart.17

### 4.2 Constructing the Sparse Design Matrix (**$X$**)

Replicating RAPM requires constructing a massive, sparse design matrix. For a single season, this matrix may have dimensions of roughly $50,000 \text{ rows} \times 1,200 \text{ columns}$.

Encoding Specification:

For a given stint where Team A (Home) is on offense:

* **Home Players (Offense):** Encoded as 1.
* **Away Players (Defense):** Encoded as -1.
* **Goalies:**
  + For **xG RAPM:** Goalies are typically *excluded* (all 0s) because xG is a skater-driven metric independent of save ability.
  + For **Goals RAPM:** Goalies are *included* (encoded as -1 for the defending goalie) to separate defensive skater play from goaltender performance.6
* **Contextual Variables:**
  + ZoneStart\_Off: 1 if stint started in Off Zone.
  + ZoneStart\_Def: 1 if stint started in Def Zone.
  + ScoreState: Dummy variables for score differential (e.g., HomeLead\_2).

### 4.3 Lambda Optimization via Cross-Validation

The choice of $\lambda$ dictates the amount of "shrinkage." A $\lambda$ of 0 is OLS (noisy); a $\lambda$ of $\infty$ sets all ratings to zero.

**Replication Protocol:**

1. Use **10-fold Cross-Validation**.
2. Split the stint data into 10 random partitions.
3. Train the Ridge model on 9 partitions with a range of $\lambda$ values.
4. Test on the 10th partition and calculate Mean Squared Error (MSE).
5. Select $\lambda\_{min}$ (the value minimizing MSE) or $\lambda\_{1se}$ (one standard error deviation for a more parsimonious model).
6. *Software Implementation:* In R, use cv.glmnet(alpha=0); in Python, use sklearn.linear\_model.RidgeCV.17

## 5. The Aggregation Engine: Wins Above Replacement (WAR)

Wins Above Replacement (WAR) is not a single model but an aggregation of multiple RAPM outputs converted into a currency of "Wins." It answers the holistic question of value by summing contributions across all facets of the game.

### 5.1 The Component Architecture

TopDownHockey and Evolving Hockey construct WAR from six isolated Ridge Regressions. To replicate this, one must run independent RAPM models for each of the following targets 19:

**Table 2: WAR Component Regression Configuration**

| **Component** | **Target Variable** | **Predictors Included** | **Filtering Condition** |
| --- | --- | --- | --- |
| **EV Offense** | xGoals For / 60 | Skaters (1), Context | Strength = 5v5 |
| **EV Defense** | xGoals Against / 60 | Skaters (-1), Context | Strength = 5v5 |
| **PP Offense** | xGoals For / 60 | Skaters (1), Context | Strength = 5v4, 5v3 |
| **PK Defense** | xGoals Against / 60 | Skaters (-1), Context | Strength = 4v5, 3v5 |
| **Penalties** | Net Penalties / 60 | Skaters (1), Context | All Strengths |
| **Finishing** | (Goals - xGoals) / 60 | Skaters (1) | All Strengths |

*Note: The "Finishing" component isolates the shooter's ability to exceed the model's expectation, distinguishing "snipers" from average shooters.*

### 5.2 The "Daisy Chain" Method for Priors

Single-season RAPM can be volatile. To stabilize evaluations, WAR models employ a "Daisy Chain" approach, effectively using Bayesian priors.

**Algorithm:**

1. Run RAPM for Season $T-1$.
2. Use the resulting coefficients as the *offset* or *prior weights* for the regression of Season $T$.
3. This informs the model of a player's historical ability, requiring significant new evidence to shift the rating drastically.
4. *Replication Note:* This is crucial for accurately valuing players with limited ice time in the current season.21

### 5.3 Replacement Level Definition and Conversion

The raw output of RAPM is "Goals Above Average" (where 0 is the league mean). To get to "Above Replacement," we must define the baseline.

Replacement Level Logic:

Replacement level is defined as the caliber of player readily available to a team for minimum cost (e.g., a waiver wire pickup or AHL call-up).

* **Statistical Definition:** Evolving Hockey defines this as the performance level of a player ranked outside the top 13 Forwards or 7 Defensemen per team by Time On Ice (TOI).
* Calculation:  
    
  $$Val\_{Rep} = Val\_{Avg} - (Baseline\_{Positional} \times Minutes)$$  
    
  where $Baseline$ is the weighted average RAPM of those "replacement" players.22

Goals-to-Wins Conversion:

Finally, the net "Goals Above Replacement" (GAR) is converted to Wins (WAR) using a Pythagorean conversion factor. Empirically in the NHL, this conversion is approximately:

$$6 \text{ Goals} \approx 1 \text{ Win}$$

$$WAR = \frac{GAR}{6}$$

## 6. The Spatial Frontier: Tracking Data and Microstats

The frontier of hockey analytics moves beyond event data to tracking data (NHL EDGE) and manually charted microstats (All-Three-Zones). These datasets allow for the quantification of concepts like "Gap Control" and "Entry Denial" that RAPM cannot see.

### 6.1 Gap Control and Entry Defense

"Gap" refers to the distance between the defender and the puck carrier at the moment of a defensive zone entry. It is a proxy for the defender's ability to challenge the rush.

Mathematical Formulation for Replication:

Using NHL EDGE tracking data ($x,y$ coordinates at 25Hz):

1. **Trigger Event:** Identify timestamp $t$ where the puck crosses the blue line ($x = \pm 25$ ft in standardized coordinates).
2. **Entity Identification:** Identify the attacking Carrier ($P\_{att}$) and the nearest Defender ($P\_{def}$).
3. Euclidean Gap:  
     
   $$Gap = \sqrt{(x\_{P\_{att}} - x\_{P\_{def}})^2 + (y\_{P\_{att}} - y\_{P\_{def}})^2}$$
4. Velocity Adjustment: A static gap is misleading if the defender is flat-footed. Advanced models calculate "Effective Gap" by incorporating the defender's backward velocity vector ($v\_{def}$).  
     
   $$Gap\_{eff} = Gap\_{static} - (v\_{def} \cdot \Delta t)$$  
     
   Logic: A defender moving backward at high speed effectively "closes" the gap faster than a stationary one.23

### 6.2 Win Direction Based Events (WDBE) for Faceoffs

Traditional Faceoff Percentage is poorly correlated with winning games. The **WDBE** metric (Win Direction Based Events) re-evaluates faceoffs based on *where* the puck goes.

The Probability Model:

$$WDBE = \sum\_{i} P(\text{WinType}\_i | \text{Skill}) \times E(\text{Outcome}\_i)$$

**Replication Steps:**

1. **Classify Wins:** Divide faceoff wins into "Clean" (direct to teammate) vs. "Scrum" (puck battle).
2. **Classify Direction:** Segment the exit vector into quadrants (e.g., "Back-Center" for clean defense, "Forward" for attack).
3. **Assign Value:** Calculate the expected next event (Shot, Clear, Turnover) for each specific win type.
   * *Insight:* A "Clean Win Back" has a significantly higher correlation to subsequent shot generation than a "Scrum Win," yet standard stats treat them as identical.25

### 6.3 Expected Threat (xT) and Possession Value

Adapted from soccer analytics (Karun Singh), **Expected Threat (xT)** assigns a value to moving the puck to dangerous areas, even if no shot is taken.

**The Grid-Based Markov Model:**

1. **Discretize the Rink:** Divide the ice into an $M \times N$ grid (e.g., $16 \times 12$).
2. **Transition Matrix ($T$):** Calculate the probability of moving the puck from Zone $i$ to Zone $j$ based on historical data.
3. **Score Probability ($S$):** The probability of a shot becoming a goal from Zone $i$.
4. The Recursive Formula:  
     
   $$xT\_i = S\_i + \sum\_{j} T\_{i \to j} \times xT\_j$$  
     
   This equation solves for the "threat" of holding the puck in Zone $i$ by summing the immediate shot probability and the weighted threat of all possible next zones.
5. Player Evaluation: A player receives xT credit for completing a pass from a low-value zone to a high-value zone:  
     
   $$Credit = xT\_{end} - xT\_{start}$$  
     
   This illuminates the value of playmakers who transition the puck through the neutral zone, a skillset often invisible to xG models.27

## 7. Replication Protocols: Code and Implementation

For the researcher attempting to replicate these models, the choice of software and library is non-trivial.

### 7.1 Python for xG and Tracking

Python is the preferred environment for xG (due to XGBoost/LightGBM integration) and Tracking data (due to JSON handling).

* **Key Library:** nhl-api-py for endpoint wrapping.
* **Key Function:** pandas.json\_normalize for flattening the nested details of tracking data.
* **Geometry:** numpy for vectorizing the Euclidean distance and angle calculations across millions of rows.

### 7.2 R for RAPM and Matrix Operations

R remains dominant for RAPM due to the glmnet package's efficient handling of sparse matrices ($50k \times 1k$).

* **Key Library:** Matrix (specifically sparseMatrix) to construct the player design matrix without exhausting RAM.
* **Key Function:** cv.glmnet(x, y, weights, alpha=0) runs the Ridge Regression with cross-validation in a single optimized call.

## 8. Conclusion

The analytical architecture of the modern NHL is built upon a foundation of **Contextual Isolation**. Whether through the geometric normalization of xG, the collinearity-busting regularization of RAPM, or the spatial derivation of tracking microstats, the unifying goal is to separate the player from the chaos of the game.

Replicating these works requires not just data access, but a strict adherence to the mathematical principles outlined in this report: the penalization of variance via Ridge Regression, the non-linear feature engineering of Gradient Boosting, and the Bayesian priors of WAR. As tracking data becomes ubiquitous, the next generation of models will likely fuse these approaches, using xT and Gap Control as inputs into even more granular RAPM regressions, further refining our understanding of value in the sport.

#### Works cited

1. coreyjs/nhl-api-py: NHL API (2025/2026 Updated) - For accessing most of the NHL EDGE statistical API's, scores, schedules and more - GitHub, accessed January 13, 2026, <https://github.com/coreyjs/nhl-api-py>
2. NHL API: what data is exposed and how to analyse it with Python | by Vadzim Tashlikovich, accessed January 13, 2026, <https://medium.com/@vtashlikovich/nhl-api-what-data-is-exposed-and-how-to-analyse-it-with-python-745fcd6838c2>
3. Zmalski/NHL-API-Reference: Unofficial reference for the NHL API endpoints. - GitHub, accessed January 13, 2026, <https://github.com/Zmalski/NHL-API-Reference>
4. SimonCoulombe/nhl\_play\_by\_play: mess with the NHL data API - GitHub, accessed January 13, 2026, <https://github.com/SimonCoulombe/nhl_play_by_play>
5. Visualizing NHL Shot Location in Looker Studio | by Marc Soares - Medium, accessed January 13, 2026, <https://medium.com/@marc_soares/visualizing-nhl-shot-location-data-in-looker-studio-8a48dd5a38fa>
6. Regularized Adjusted Plus-Minus (RAPM) - Evolving-Hockey, accessed January 13, 2026, <https://evolving-hockey.com/glossary/regularized-adjusted-plus-minus/>
7. Reviving Regularized Adjusted Plus-Minus for Hockey, accessed January 13, 2026, <https://hockey-graphs.com/2019/01/14/reviving-regularized-adjusted-plus-minus-for-hockey/>
8. [MoneyPuck.com] We've made a tweak to our xGoals model to adapt to how the NHL is now tracking tips/deflections. These shots were overvalued in xG, resulting in most goalies looking above average. : r/hockey - Reddit, accessed January 13, 2026, <https://www.reddit.com/r/hockey/comments/1q3q5li/moneypuckcom_weve_made_a_tweak_to_our_xgoals/>
9. NHL Expected Goals (xG) Model - by Nick Sofianakos - Medium, accessed January 13, 2026, <https://medium.com/@nsofianakos/nhl-expected-goals-xg-model-39bd2edba932>
10. An NHL expected goals (xG) model built with light gradient boosting. - GitHub, accessed January 13, 2026, <https://github.com/JNoel71/NHL-Expected-Goals-xG-Model>
11. CARLETON UNIVERSITY SCHOOL OF MATHEMATICS AND STATISTICS HONOURS PROJECT, accessed January 13, 2026, <https://carleton.ca/math/wp-content/uploads/Kennedy-Theresa-Final-Copy-PDF.pdf>
12. Linköping Hockey Analytics Conference LINHAC 2024 - IDA.LiU.SE, accessed January 13, 2026, <https://www.ida.liu.se/research/sportsanalytics/LINHAC/LINHAC24/LINHAC2024_proceedings.pdf>
13. Evolving Hockey Overview, accessed January 13, 2026, <https://evolving-hockey.com/evolving-hockey-overview/>
14. Glossary - MoneyPuck.com, accessed January 13, 2026, <https://moneypuck.com/glossary.htm>
15. About and How it Works - MoneyPuck.com, accessed January 13, 2026, <https://moneypuck.com/about.htm>
16. Reviving Regularized Adjusted Plus-Minus for Hockey - Evolving-Hockey, accessed January 13, 2026, <https://evolving-hockey.com/blog/reviving-regularized-adjusted-plus-minus-for-hockey/>
17. How Do You Tune Lambda In Ridge Regression With Cross-validation? - The Friendly Statistician - YouTube, accessed January 13, 2026, <https://www.youtube.com/watch?v=5OgOv9yYljM>
18. Adjusted Plus-Minus for NHL Players using Ridge Regression with Goals,Shots, Fenwick, and Corsi | Request PDF - ResearchGate, accessed January 13, 2026, <https://www.researchgate.net/publication/51969146_Adjusted_Plus-Minus_for_NHL_Players_using_Ridge_Regression_with_GoalsShots_Fenwick_and_Corsi>
19. Wins Above Replacement — High Level Overview | by Patrick Bacon - Medium, accessed January 13, 2026, <https://medium.com/data-science/wins-above-replacement-high-level-overview-63838e914395>
20. What is NHL WAR? Wins Above Replacement Explained | HockeyStats.com, accessed January 13, 2026, <https://hockeystats.com/methodology/war>
21. Wins Above Replacement 1.1 and Expected Goals 1.1: Model Updates and Validation | by Patrick Bacon | TDS Archive | Medium, accessed January 13, 2026, <https://medium.com/data-science/wins-above-replacement-1-1-and-expected-goals-1-1-model-updates-and-validation-c05855b59f12>
22. Goals Above Replacement (GAR - Evolving-Hockey, accessed January 13, 2026, <https://evolving-hockey.com/glossary/goals-above-replacement/>
23. The Analytics Don't Share The Optimism : r/Habs - Reddit, accessed January 13, 2026, <https://www.reddit.com/r/Habs/comments/1o3elkc/the_analytics_dont_share_the_optimism/>
24. Philadelphia Flyers 2015-16 Full-Season Zone Entry Tracking Data and Observations, accessed January 13, 2026, <https://www.broadstreethockey.com/post/philadelphia-flyers-zone-entry-tracking-data-and-observations-full/>
25. The puck drops here | Smith Magazine, accessed January 13, 2026, <https://smith.queensu.ca/magazine/issues/summer-2019/profiles/puck-drops-here.php>
26. (PDF) Winning Is Not Everything: A contextual analysis of hockey face-offs - ResearchGate, accessed January 13, 2026, <https://www.researchgate.net/publication/330955747_Winning_Is_Not_Everything_A_contextual_analysis_of_hockey_face-offs>
27. Dynamic Expected Threat (DxT) Model: Addressing the Deficit of Realism in Football Action Evaluation - MDPI, accessed January 13, 2026, <https://www.mdpi.com/2076-3417/15/8/4151>
28. Expected Total Threat - Off The Wall Analytics - WordPress.com, accessed January 13, 2026, <https://offthewallanalytics.wordpress.com/2021/01/18/expected-total-threat/>